METHOD AND APPARATUS FOR AVOIDANCE OF ROUTE-DESIGNATION DEADLOCK OF PACKET

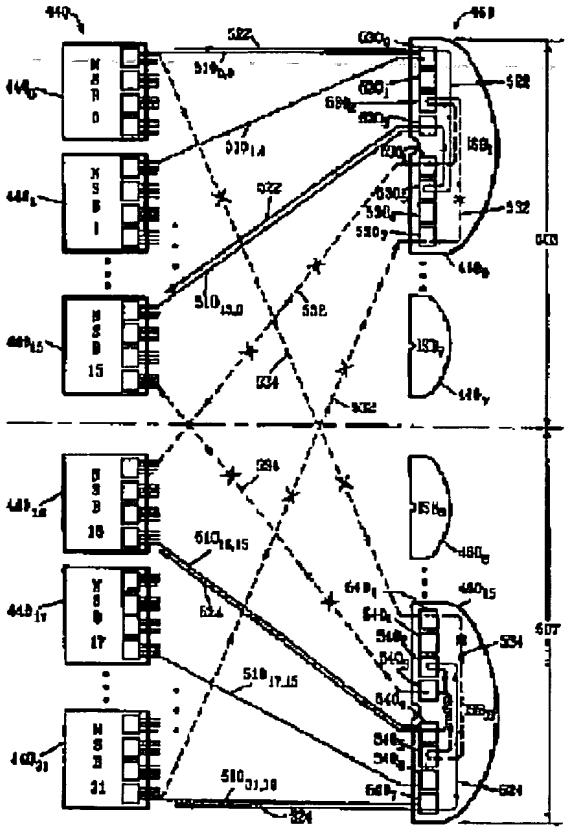
Patent number: JP7282018
Publication date: 1995-10-27
Inventor: HARISHI SESU; ROBAATO FUREDERITSUKU SUTATSUK; KUREIGU BURAIAN SUTANKERU
Applicant: IBM
Classification:
- international: G06F15/16; G06F15/173
- european: H04L12/56C
Application number: JP19950012797 19950130
Priority number(s): US19940222284 19940404

Also published as:
EP0676703 (A2)
US5453978 (A1)
EP0676703 (A3)
EP0676703 (B1)

Report a data error here

Abstract of **JP7282018**

PURPOSE: To provide a device and method to establish path designation without a deadlock in a large scale 2-way multi-stage interconnecting cross point switch base packet network. CONSTITUTION: In the case of selecting a path included in a path table in a system, the entire network is effectively separated so as to inhibit specific paths to separate the system into prescribed divisions, e.g. to separate a packet traffic mostly flowing among nodes of a half part 503 of the system from a packet traffic flowing among nodes of the other half part 507 in the system. In order to extract paths of a packet passing among nodes in a common division of the system from this standpoint, paths including paths 522, 524 passing through the other system division are inhibited. The path inhibit as above is not caused in the selection of paths through which a packet is propagated among nodes in a plurality of the system divisions, e.g. nodes included in different halves of the system.



Data supplied from the *esp@cenet* database - Worldwide

BEST AVAILABLE COPY

(19)日本国特許庁(JP)　(12)公開特許公報（Ａ）　(11)特許出願公開番号

特開平７－２８２０１８

(43)公開日　平成7年(1995)10月27日

(51)Int.Cl.[6]	識別記号	庁内整理番号	ＦＩ	技術表示箇所
Ｇ０６Ｆ　15/16	４７０　Ａ			
15/173				
			Ｇ０６Ｆ　15/16　４００　Ｎ	

審査請求　有　請求項の数２０　ＯＬ　（全１８頁）

(21)出願番号　特願平7-12797

(22)出願日　平成7年(1995)1月30日

(31)優先権主張番号　222284
(32)優先日　1994年4月4日
(33)優先権主張国　米国（ＵＳ）

(71)出願人　390009531
インターナショナル・ビジネス・マシーンズ・コーポレイション
ＩＮＴＥＲＮＡＴＩＯＮＡＬ　ＢＵＳＩＮＥＳＳ　ＭＡＳＣＨＩＮＥＳ　ＣＯＲＰＯＲＡＴＩＯＮ
アメリカ合衆国10504、ニューヨーク州　アーモンク　（番地なし）

(72)発明者　ハリシ・セス
アメリカ合衆国12401、ニューヨーク州キングストン、ナンバー310、ウィルバー・アベニュー　162－182

(74)代理人　弁理士　合田　潔　（外2名）

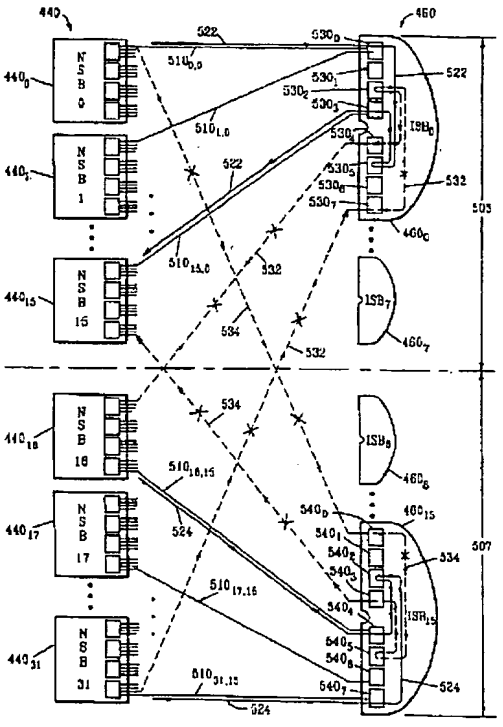
最終頁に続く

(54)【発明の名称】パケットの経路指定デッドロック回避方法及び装置

(57)【要約】

【目的】　大規模双方向マルチステージ相互接続クロスポイント・スイッチ・ベース・パケット・ネットワークにおけるデッドロックの無い経路指定を確立する装置及び方法を提供する。

【構成】　システム内の経路テーブルに含まれる経路の選択において、システムのある区分、例えばシステムの半分５０３内のノード間をもっぱら流れるパケット・トラフィックを、他の区分、例えばシステムの別の半分５０７内のノード間を流れるパケット・トラフィックから分離するために、特定の経路を禁止するように、ネットワーク全体が効果的に区分される。この点に関し、システムの共通区分内のノード間を通過するパケットの経路を抽出するために、他のシステム区分を通過するパス５２４、５４４を含む経路が禁止される。複数のシステム区分、例えばシステムの異なる半分に含まれるノード間でパケットを伝搬する経路の選択においては、こうした経路の禁止は発生しない。



1

【特許請求の範囲】

【請求項１】パケット・ネットワークの外部の複数のノードを集合的に相互接続するクロスポイント・スイッチの連続ステージを含む前記ネットワークを有する装置において、パケットが前記ネットワーク及び少なくとも１つの前記スイッチを介して、規定経路上を第１の前記ノードから第２の前記ノードに伝搬されるものにおいて、前記ネットワーク内における経路指定デッドロックの発生を回避する実質的な方法であって、

パケットが前記複数のノード内の個々のノードから、異なる対応する前記経路を介して、前記複数のノードのあらゆる他のノードに伝搬されるように、前記ネットワークを介する複数の規定経路を第１に定義するステップであって、前記の各定義経路が少なくとも１つのリンクに伸び、第１のネットワーク区分だけに接続される第１及び第２の前記ノード間を通過するパケットが、第２のネットワーク区分に伸びるリンクを有する経路上で伝搬されないように、前記ネットワークを前記第１及び前記第２のネットワーク区分に分割するように前記規定経路を定義する、前記第１の定義ステップと、

全ての前記規定経路を結果の経路テーブルに記憶するステップと、

を含む、方法。

【請求項２】前記第１の定義ステップが、前記第１及び前記第２のネットワーク区分にそれぞれ接続される第３及び第４のノード間を通過するパケットが、前記第１及び前記第２のネットワーク区分間に伸びる少なくとも１つのリンクを有する経路上で伝搬されるように、前記複数の規定経路を定義する第２の定義ステップを含む、請求項１記載の方法。

【請求項３】前記ネットワーク内において出所ノードから宛先ノードに経路指定されるパケットをアセンブルする際に、前記第３及び前記第４の両ノード内において、

前記パケットの結果的経路を生成するために、前記経路テーブルをアクセスするステップと、

前記結果的経路を前記パケットにコピーするステップと、

前記パケットを前記結果的経路上で前記ネットワークを介して経路指定するステップと、

を含む、請求項２記載の方法。

【請求項４】前記経路テーブルの異なる部分を、前記複数の各ノードに対応する別々の局所経路テーブルにダウンロードするステップであって、前記の各経路テーブル部分が前記各ノードを出所ノードとして有する全ての前記規定経路を指定する、前記ダウンロード・ステップを含み、

前記経路コピー・ステップが、前記出所ノードから前記宛先ノードに伝搬されるパケットの前記結果的経路を生成するために、前記パケットの前記宛先ノードにもとづき、前記出所ノードの前記局所経路テーブルをアクセスするステップを含む、請求項３記載の方法。

2

【請求項５】前記の各パケットが少なくとも１つの経路バイトを含む経路フィールドを有するヘッダを含み、前記経路フィールドが前記各パケットが前記ネットワークを伝わる経路を集合的に指定し、各個々の前記経路バイトが前記各パケットが対応する前記クロスポイント・スイッチの１つを横断する経路を定義し、前記結果的経路のコピー・ステップが前記結果的経路内の各連続する前記経路バイトの値を前記ヘッダ内の別々の対応する連続経路バイトにコピーするステップを含む、請求項４記載の方法。

【請求項６】各ネットワーク区分が前記ネットワークの異なる半分を構成する、請求項５記載の方法。

【請求項７】前記装置をサービス・フェーズ及び実行フェーズで動作させるステップであって、前記第１の定義ステップ及び前記規定経路記憶ステップを前記サービス・フェーズの間に実行し、前記結果的経路アクセス・ステップ、前記結果的経路コピー・ステップ及び前記パケット経路指定ステップを前記実行フェーズの間に実行する前記動作ステップを含む、請求項５記載の方法。

【請求項８】前記第１の定義ステップが、

トポロジ・ファイル内のネットワーク装置及び相互接続データに応答して、出所ノードとしての前記の各ノードから、宛先ノードとしてのあらゆる他の使用可能な前記ノードの１つへの全ての使用可能な最短パス経路を決定するステップであって、前記トポロジ・ファイル内に含まれるある前記装置に対するデッドロック回避指示により禁止される前記装置を通過するパスを有する経路を、前記最短パス経路から除外する、前記決定ステップと、

ある前記出所ノードとある前記宛先ノード間で１つの前記最短パス経路が存在する場合、前記最短パス経路を前記経路テーブルに、前記出所ノードと前記宛先ノード間の規定経路として書込むステップと、

ある前記出所ノードとある前記宛先ノード間で複数の最短パス経路が存在する場合、前記最短パス経路の中から、集合的に最小の重みを有する１つの前記最短パス経路を、前記出所ノードと前記宛先ノード間の前記規定経路として選択するステップと、

前記規定経路内の各リンクに対応する別々の重みを、予め定義された量だけ増分するステップと、

を含む、請求項５記載の方法。

【請求項９】前記全ての使用可能な経路の決定ステップが、前記全ての使用可能な最短パス経路を突き止めるブレッドス・ファースト探索を実行するステップを含む、請求項８記載の方法。

【請求項１０】前記装置をサービス・フェーズ及び実行フェーズで動作させるステップであって、前記第１の定義ステップ及び前記規定経路記憶ステップを前記サービス・フェーズの間に実行し、前記結果的経路アクセス・ステップ、前記結果的経路コピー・ステップ及び前記パ

ケット経路指定ステップを前記実行フェーズの間に実行する前記動作ステップを含む、請求項９記載の方法。

【請求項１１】各ネットワーク区分が前記ネットワークの異なる半分を構成する、請求項１０記載の方法。

【請求項１２】パケット・ネットワークの外部の複数のノードを集合的に相互接続するクロスポイント・スイッチの連続ステージを含む前記ネットワークを有するシステムにおいて、パケットが前記ネットワーク及び少なくとも１つの前記スイッチを介して、規定経路上を第１の前記ノードから第２の前記ノードに伝搬されるものにおいて、前記ネットワーク内における経路指定デッドロックの発生を回避する装置であって、

パケットが前記複数のノード内の個々のノードから、異なる対応する前記経路を介して、前記複数のノードのあらゆる他のノードに伝搬されるように、前記ネットワークを介する複数の規定経路を定義する第１の手段であって、前記の各定義経路が少なくとも１つのリンクに伸び、第１のネットワーク区分だけに接続される第１及び第２の前記ノード間を通過するパケットが、第２のネットワーク区分に伸びるリンクを有する経路上で伝搬されないように、前記ネットワークを前記第１及び前記第２のネットワーク区分に分割するように前記規定経路を定義する、前記第１の定義手段と、

全ての前記規定経路を結果の経路テーブルに記憶する手段と、

を含む装置。

【請求項１３】前記第１の定義手段が、前記第１及び前記第２のネットワーク区分にそれぞれ接続される第３及び第４のノード間を通過するパケットが、前記第１及び前記第２のネットワーク区分間に伸びる少なくとも１つのリンクを有する経路上で伝搬されるように、前記複数の規定経路を定義する、請求項１２記載の装置。

【請求項１４】前記ネットワーク内において出所ノードから宛先ノードに経路指定されるパケットをアセンブルする間に、前記第３及び前記第４の両ノード内において、前記パケットの結果的経路を生成するために、前記経路テーブルをアクセスし、前記結果的経路を前記パケットにコピーし、前記パケットを前記結果的経路上で前記ネットワークを介して経路指定する手段を含む、請求項１３記載の装置。

【請求項１５】前記経路テーブルの異なる部分がダウンロードされる前記複数の各ノードに対応する別々の局所経路テーブルであって、前記の各経路テーブル部分が前記各ノードを出所ノードとして有する全ての前記規定経路を指定する、前記局所経路テーブルと、

前記出所ノードから前記宛先ノードに伝搬されるパケットの前記結果的経路を生成するために、前記パケットの前記宛先ノードにもとづき、前記出所ノードの前記局所経路テーブルをアクセスする手段と、

を含む、請求項１４記載の装置。

【請求項１６】前記の各パケットが少なくとも１つの経路バイトを含む経路フィールドを有するヘッダを含み、前記経路フィールドが前記各パケットが前記ネットワークを伝わる経路を集合的に指定し、各個々の前記経路バイトが前記各パケットが対応する前記クロスポイント・スイッチの１つを横断する経路を定義し、前記結果的経路内の各連続する前記経路バイトの値を、前記ヘッダ内の別々の対応する連続経路バイトにコピーする、請求項１５記載の装置。

【請求項１７】各ネットワーク区分が前記ネットワークの異なる半分を構成する、請求項１６記載の装置。

【請求項１８】前記第１の定義手段が、

トポロジ・ファイル内のネットワーク装置及び相互接続データに応答して、出所ノードとしての前記の各ノードから、宛先ノードとしてのあらゆる他の使用可能な前記ノードの１つへの全ての使用可能な最短パス経路を決定する手段であって、前記トポロジ・ファイル内に含まれるある前記装置に対するデッドロック回避指示により禁止される前記装置を通過するパスを有する経路を、前記最短パス経路から除外する、前記決定手段と、

ある前記出所ノードとある前記宛先ノード間で１つの前記最短パス経路が存在する場合、前記最短パス経路を前記経路テーブルに、前記出所ノードと前記宛先ノード間の規定経路として書込む手段と、

ある前記出所ノードとある前記宛先ノード間で複数の最短パス経路が存在する場合、前記最短パス経路の中から、集合的に最小の重みを有する１つの前記最短パス経路を、前記出所ノードと前記宛先ノード間の前記規定経路として選択する手段と、

前記規定経路内の各リンクに対応する別々の重みを、予め定義された量だけ増分する手段と、

を含む、請求項１６記載の装置。

【請求項１９】前記システムが並列処理システムであり、前記の各ノードが別々の処理要素を含む、請求項１６記載の装置。

【請求項２０】前記並列処理システムが５１２個の別々の処理要素を含み、前記スイッチが３２ポート・スイッチ・ボードに編成され、前記システムが３２個のノード・スイッチ・ボード（ＮＳＢ）と１６個の中間スイッチ・ボード（ＩＳＢ）による複数のスイッチ・ボードを含み、前記ＩＳＢが、前記の各ＮＳＢ上の１６ポートのそれぞれが、異なる対応するリンクを介して、前記の各ＮＳＢ上の同一の対応するポートに接続されるように、また前記の各ＮＳＢ上の残りの１６ポートが１６個の異なる連続する前記処理要素に接続されるように、全ての前記ＮＳＢを集合的に相互接続する、請求項１９記載の装置。

【発明の詳細な説明】

【０００１】

【産業上の利用分野】本発明はマルチステージ相互接続

クロスポイント・ベースのパケット交換を確立するための装置及び方法に関する。特に本発明は、大容量並列処理システム内で使用される高速パケット・ネットワーク内に組込むのに適するが、それに限るものではない。

【０００２】

【従来の技術】強力で知能的で比較的安価なマイクロプロセッサの継続的な進歩及び市場での可用性により、大容量並列処理が、これまで従来式のメインフレーム・コンピュータにより処理されてきた広範なアプリケーション、例えばトランザクション処理、シミュレーション及び構造解析などを処理するために、益々魅力的な手段に成りつつある。

【０００３】大容量並列処理システムでは、しばしば数百または数千にも上る相当数の比較的単純なマイクロプロセッサを基本とする別々の処理要素が、一般に高速パケット・ネットワークから形成される通信構造を介して相互接続され、各こうした処理要素がネットワーク上の別々のポートとして現れる。この構造はパケット形式の構造経路メッセージを、これらの処理要素の任意の１つから他へ経路指定し、それらの間の通信を提供する。これらの各々の処理要素は、通常、別々のマイクロプロセッサ及びその関連支援回路を含み、後者はとりわけ、一時記憶用のランダム・アクセス・メモリ（ＲＡＭ）及び永久記憶用の読出し専用メモリ（ＲＯＭ）、及び入出力回路により代表される。各処理要素は更に通信サブシステムを含み、これは適切な通信インタフェース及び他のハードウェア、並びにこの要素をパケット・ネットワークにインタフェースするように集合的に機能する制御ソフトウェアにより形成される。

【０００４】一般に、大容量並列処理システムの全体性能は、そこで使用される根元的なパケット・ネットワークの性能により制限される。その点でパケット・ネットワークが余りに遅く、特に全体システム・スループットに悪影響を及ぼす程度に遅いと、結果的な低下は著しく、所与のアプリケーションにおいて大容量並列処理システムを使用する効果が低減する。

【０００５】特に、大容量並列処理システムにおいて、各処理要素はアプリケーションの予め定められた細分化部分を実行する。その対応するアプリケーション部分の実行において、各要素は一般に、例えば異なる要素上で実行されるアプリケーション部分からデータを要求し、処理結果データを例えば、更に別の処理要素上で実行される別のアプリケーション部分に提供する。全ての要素間における処理の相互依存の性質により、各処理要素は、その時これらの各々の要素において実行されるアプリケーション部分からの要求により、データを別のこうした要素に転送できなければならない。一般に処理要素、例えば”宛先”要素が、別のこうした要素、例えば”出所”要素または”発信元”要素に対してデータを要求すると、宛先要素は少なくともこの特定のアプリケーション部分に関し、その要素が出所要素により伝送される必要データを含むパケットを受信するまで遊休状態を維持する。パケットを受信すると、宛先要素は再度このアプリケーション部分の処理を開始する。パケット・ネットワークを通じて宛先からの要求を含むパケットを出所処理要素に移送し、次に要求データを含む応答パケットを反対方向に移送するためには、有限量の時間が必要である。この時間は、宛先要素において実行されるそのアプリケーション部分に、ある程度の待ち時間を不可避に挿入する。システム内のほとんどの処理要素が、出所要素において実行されるアプリケーション部分に対応する宛先要素として機能するので、この通信により誘導される待ち時間が余りに長いと、システム・スループットが顕著に低下する。結果的に、このことは全体システム性能を著しく低下させることになる。これを回避するためにパケット・ネットワークは各パケットを任意の２つの通信処理要素間で、この待ち時間を低減するように、可能な限り速く移送しなければならない。更に、典型的な大容量並列処理システムにおいて一般に使用される相当数の処理要素、及びこのシステム内の任意のある要素が、任意の時刻において、他のこうした要素と通信するために必要な付随のニーズを考慮すると、ネットワークは相当に大きな数、例えば予測されるピーク負荷のパケットを処理要素間で同時に経路指定できなければならない。

【０００６】しかしながら、実際には、大容量並列処理システムで使用される必要性能、特に伝送帯域幅を所有するパケット交換ネットワークは、様々な理由から、その開発が極めて困難であることがわかっており、そのためにこうしたシステムの急速な普及及び使用の増加がある程度阻止されてきた。

【０００７】パケット・ネットワークの様々な形態が今日存在するが、１つの共通のアーキテクチャとしては、比較的小さなクロスポイント・スイッチのマルチステージ相互接続構成を使用する。各スイッチは、通常、８ポート双方向ルータであり、全てのポートがクロスポイント・マトリックスを通じて内部的に相互接続される。こうしたネットワークでは、１ステージ内の各スイッチはネットワークの片側（すなわち、いわゆる”入力”）において開始し、特定の対応するパス（典型的にはバイト幅物理接続）を通じて、次の続くステージ内のスイッチに相互接続され、このようにして、ネットワークの反対側（すなわち、いわゆる”出力”）の最後のステージに到達するまで継続される。こうしたスイッチは、今日、動作的には非ブロッキング（non-blocking）の比較的安価な単一の集積回路（以降では”スイッチ・チップ”として参照される）として調達可能であるので、これらのスイッチ・チップが好まれて使用される。実際に、中央キューの使用に頼る非ブロッキング８ウェイ・ルータとして実現されるこうしたスイッチ・チップが、P. Hochschildらによる係属中の米国特許出願第０２７９０６号”A Cen

tral Shared QueueBased Time Multiplexed Packet Switch with Deadlock Avoidance”（１９９３年３月４日出願）に述べられている（本願の出願人に権利譲渡される）。

【０００８】こうした双方向マルチステージ・パケット交換ネットワークは、他のパケット交換ネットワーク・トポロジと比較して比較的単純であり、その全てのポート間で高い伝送帯域幅を提供するが、残念ながらこのタイプのネットワークは、経路指定デッドロックを受け易い。これらのデッドロックは稀にしか発生しないが、同一ステージ内の任意の２つのスイッチ間に複数の経路が存在するために、実際に発生する。

【０００９】この点に関し、８個のこうしたスイッチ・チップが２つの相互接続ステージに編成される単純な３２ポート・ネットワークについて考えてみよう。すなわち、４個のスイッチによる入力ステージの後に、４個のスイッチによる出力ステージが続き、これらの全てのスイッチ・チップが単一のスイッチ・ボード上に含まれる。この構成では、入力ステージにおいて、異なるスイッチ・チップ上の任意の２つのポート間を通過するパケットは、出所（”入力”）ポートを含む入力ステージ内のスイッチ・チップを通過して、出力ステージの４個のスイッチ・チップの１個に経路指定される。次に、この後者のスイッチ・チップが、パケットをこのパケットの宛先（”出力”）ポートを含む入力ステージ内のスイッチに逆経路指定する（すなわち、その方向を反転する）。スイッチ・チップ間の経路は、通常、比較的短い時間に渡り、各バイト・ワイズ（byte-wise）・パスがほぼ等しい数のパケットを伝搬し、ネットワーク全体を通じてのトラフィック・フローを平均化するように、システム初期化の間に予め定義される。これらの経路が設定されると、スイッチ・チップまたはパス故障或いは保守状態以外では経路は稀にしか変更されない。各処理要素が使用可能な割当てられた経路が、次に再度システム初期化の間に（局所）経路テーブルの形式でその要素に提供される。引続きルーチンのオペレーションの間に、各処理要素がパケットを形成すると、その要素はこのパケットの宛先にもとづきその経路テーブルから経路を読出し、単にその経路をパケットのヘッダ内に適切な経路バイトの値として挿入する。パケットが次にネットワーク内に送出され、パケット内の対応する経路バイトの値により指定される継続するスイッチ・チップ（及び交換ステージ）を経由して、経路指定される。パケットが交換ステージを経由して横断すると（すなわち、ここでは同一ステージの２個のスイッチ・チップを通過する）、ステージ内の最後のスイッチ・チップがパケット・ヘッダから対応する経路バイトを切捨てる。

【００１０】経路は従来、経路指定デッドロックの潜在性を考慮すること無く定義されてきた。従って、各々が例えば異なるスイッチ・チップのグループの中央キュー内に存在する対応するパケットが、連続ステージ内のスイッチ・チップ対を接続する共通パス上を同時に経路指定されるのを待機する度に、経路指定デッドロックが発生する。こうした状態が発生すると、これらの各々のスイッチ・チップは、グループ内の他のスイッチ・チップがそれらのパケットをこれらの特定のパス上に経路指定するのを待機する。このグループのどのパケットも、このグループの任意の１つのパケットが経路指定されるまで、その関連する中央キューを通過することができないので、これら全てのパケットがひたすら待機し、対応するパスがデッドロック状態となり、その上をトラフィック・フローが生じなくなる。その結果、デッドロックが発生すると、これらのパケットが宛先指定される処理要素についても、これらのパケットを待機し続けることになり、それらの処理のスループットを停止させる。結果的にネットワークの帯域幅はデッドロックにより影響されない残りの処理要素だけを優遇するようになり、処理の作業負荷が著しく偏り、システム・スループットを多大に低下させることになる。

【００１１】デッドロックを回避する問題に直面して、当業者は最初に経路指定デッドロックを予測するために、特定のタイプの大域アービトレーション手法が使用可能であると考え、多数の非デッドロック状態のパスのいずれかを選択し、その上でパケットを伝送し、デッドロックを回避することを期待するであろう。この手法は、潜在的な経路指定デッドロックを検出し、それに従い調停するために全ての中央キューを通過する全てのパケットがモニタされることを必要とする。残念ながら、これらの機能を達成する回路は極めて複雑であり、全ての各スイッチ回路の外部に配置されて、それらの各々と接続される必要がある。これはパケット交換ネットワークのサイズ、複雑度、従ってコストを押し上げることになる。この手法自体は、極めて非現実的である。

【００１２】こうしたことを考慮して、当業者は２重のスイッチ・ボードを有するパケット・ネットワークを形成するなどの別の手法に注目するであろう。この手法を３２プロセッサ・システムと共に使用することにより、１つのスイッチ・ボードのポート１６乃至３１で表される１６個のポートが、別のスイッチ・ボードの同じポートに接続される。両方のボード上の残りの各ポート０乃至１５は、３２個の別々の処理要素の対応する１つに接続される。オペレーションにおいて、共通のスイッチ・ボードに接続される出所ポートと宛先ポートとの間を通過するパケットが、もっぱらその１つのスイッチ・ボード内に経路指定され、他のスイッチ・ボード内に含まれるスイッチ・チップに影響を及ぼすことはない。異なるスイッチ・ボード上の出所ポートと宛先ポートとの間を経路指定されるパケットだけがボード間を経路指定される。片方のスイッチ・ボード内だけを流れるパケットを他のスイッチ・ボード内だけを同時に流れるパケットと

潜在的に相互作用しないように分離することにより、この手法はデッドロックを排除する。更にこの手法は伝送帯域幅を悪化させない。残念なことにこの手法は２重のスイッチ・ボード及び関連回路を必要とすることにより高価である。それにも関わらず、スイッチ・ボード及び関連回路を２重化する追加のコストが３２プロセッサ・システムにおいては許容可能である。この手法自体が、３２プロセッサ・システムにおけるデッドロックを回避するために使用される。実際に３２プロセッサ・システムでは、１つのスイッチ・ボードだけではパケット・ネットワークの形成を妨げる十分なデッドロックの潜在性が存在する。しかしながら、このコスト的な欠点は、例えば５１２プロセッサ・システムなどのように、ネットワーク内で必要とされる最小１６個のスイッチボードに加え、追加の１６個のスイッチ・ボードを必要とする大規模システムの場合に、高額でより付けないに過ぎない。

【００１３】最後に、当業者は、特定の経路の使用を単に禁止することにより、経路指定デッドロックを回避する手法を考慮するであろう。この特定の手法により、同一ステージ内の２個のスイッチ・チップ間の全ての経路の特定のサブセットだけが、それらの間のパケット・トラフィックの伝搬に使用可能と定義され、経路テーブル内に含まれる。１度選択されると、これらの経路は保守状態または故障状態以外では変化しない。サブセットを形成する経路は、特に経路指定デッドロックが発生しないように選択される。各追加の経路が禁止されるとネットワーク帯域幅が低下するので、この手法の目標はできる限り少ない経路を禁止することである。

【００１４】しかし残念ながら、経路が禁止されると”禁止されない”経路がシステム内の全てのノードに関して、対称でないことが知られている。その結果、伝送帯域幅がネットワーク全体に渡り均等に低減されず、ネットワーク全体に渡って帯域幅の非対称が生じる。これらの非対称の結果、ネットワークは伝送帯域幅が特定の”ホット”・ポートにおいて非常に高くなる傾向にあり、他では実質的に０となる、いわゆる”ホット・スポット”を発展させる傾向を示す。これは次に、他のポートを犠牲にして”ホット”・ポートに関連する処理要素を優遇するように処理スループットを偏らせ、ネットワーク全体に渡る作業負荷処理の平衡を失わせる。結果的にシステム性能の低下が生じる。実際に、経路がもっぱらスイッチ・ボード内で禁止されると、ネットワーク全体に渡り一定の帯域幅の低下をもたらす残りの禁止されない経路の任意の組合わせを見い出すことができないことが判明した。

【００１５】経路を禁止する手法は、単に各処理要素に対応する経路テーブル内に含むための特定のエントリの選択を要求するだけなので、この手法は、具体化が非常に単純で高度にコスト有効である。従ってこの手法は、ネットワーク全体に渡り対称的な帯域幅の低下を生成不能でない限り、マルチステージ・クロスポイント・パケット・ネットワークに取り入れられることが望まれる。

【００１６】相互接続双方向マルチステージ・クロスポイント・ベースのネットワークを、大容量並列処理システムの通信中枢として使用する関心にも関わらず、これらのネットワークにおけるデッドロックの潜在性の増加、並びに特に大規模ネットワークにおける現実的なソルーションの欠如が少なくとも今日まで、３２をはるかに越えるプロセッサを有するこうしたネットワークを使用する大容量並列処理システムの市場での使用可能性を抑制してきており、特定の大規模処理アプリケーションにおけるこれらのシステムの使用を妨げてきた。

【００１７】従って、大規模双方向マルチステージ相互接続クロスポイント交換ネットワークにおいて、特に、大規模大容量並列処理システムにおいて、デッドロックの発生を防止する現実的な手法が必要とされる。こうした手法は具体化が単純であり、高度にコスト有効であるべきであり、ネットワーク帯域幅が結果的に低減される場合、ネットワーク全体に渡り、実質的に対称で受諾可能なレベルの帯域幅の低減を提供するべきである。こうした手法がこうしたシステム内に含まれると、これらのシステムは市販されると３２をはるかに上回る、例えば５１２或いはそれ以上の別々のプロセッサに拡張される。従って、こうしたシステムは従来不可能であった追加のアプリケーション処理のニーズに応えることができる。

【００１８】

【発明が解決しようとする課題】本発明により、大規模双方向マルチステージ相互接続クロスポイント・スイッチ・ベース・パケット・ネットワークにおいて、経路指定デッドロックの発生を防止するための、従来技術に固有の欠点を有利に克服する単純でコスト有効な手法が提供される。この手法は、理想的には大規模大容量並列処理システムの通信中枢を形成するこうしたパケット・ネットワークにおいて使用される。

【００１９】

【課題を解決するための手段】特に本発明により、特定の予め定義された経路が経路テーブルの形成の間に、これらのノードを使用するネットワーク内における特定のノード、例えば処理要素の相対ロケーションにもとづき考慮から除外される。禁止された経路は、もしそれらが使用されなければ、閉ループ経路指定パターン、従って経路指定デッドロックの発生を防止する経路として選択される。経路テーブルに含まれる経路の選択においてシステムのある区分、例えばシステムの半分内のノード間をもっぱら流れるパケット・トラフィックを別の区分、例えばシステムの他の半分内のノード間を流れるパケット・トラフィックから分離するために経路が禁止される。この点に関し、システムの共通区分内のノード間を

通過するパケットの経路を抽出するために、システムの別の区分を通過するパス（ケーブルなど）を含む経路が禁止される。複数のシステム区分、例えばシステムの異なる半分に含まれるノード間でパケットを伝搬する経路の選択においては、こうした経路の禁止は発生しない。

【００２０】例えば、８×８のスイッチ回路（ここでは”スイッチ・チップ”としても参照される）の使用において、多数の同一の３２０ポート・スイッチ・ボードを有する５１２プロセッサ・システムが構成され、これらが２つの相互接続ステージ、すなわち、個々の処理要素に接続されるノード・スイッチ・ボード（ＮＳＢ）及びノード・スイッチ・ボード自身を相互接続するために使用される中間スイッチ・ボード（ＩＳＢ）に編成される。各ＮＳＢは１６個のそれぞれ異なる処理要素に接続される１６ポートと、１６個の各ＩＳＢ上の異なるポートに相互接続される別の１６ポートを提供する。

【００２１】このシステムにおいて禁止される経路を決定するために、１６個の連続的なＮＳＢ（例えばＮＳＢ０乃至１５、及び１６乃至３１）及び２５６個の連続的な処理要素が各半分を構成するように、システムが半分に分割されて効果的に考慮される。第１の８個のＩＳＢが一方の半分に含まれ、残りの８個のＩＳＢが他の半分に含まれる。システムの共通の半分内に配置される処理要素間を通過するパケットに対して、システムのその半分内に完全に含まれるＩＳＢポートを含む使用可能経路だけが許可され他の経路は禁止される。従って、システム初期化の間に、後者の任意の経路は、これらの処理要素を接続する大域経路テーブル内に含まれない。或いはシステムの異なる半分内に配置される処理ノード間を通過するパケットに対しては、こうした経路は禁止されない。従ってこの場合には、大域経路テーブル内に結果的に含まれる経路の選択は、システムの半分にもとづく制限無しに、使用可能な全ての経路の中から実施される。

【００２２】システムの各区分例えば半分を、他の任意の区分例えば他の半分内に含まれる処理要素対間をもっぱら流れるパケット・トラフィックから分離することにより、これらのパケットの相互作用により生じる経路指定デッドロックが有利に防止される。これにより市販の並列処理システムは、より多くの処理要素を含むように容易に拡張され、従来可能であった以上に広範な様々なアプリケーション処理ニーズに応えることができる。

【００２３】

【実施例】当業者には容易に理解されるように、双方向マルチステージ相互接続クロスポイント・ベース・パケット・スイッチを含むパケット・ネットワークは、それらの特定のアプリケーションに関係無く、ここで指摘されるタイプの経路指定デッドロックの影響を受け易い。従って次の説明を考慮した後に、当業者においては、本発明の教示がほとんどのこうしたパケット・ネットワークに容易且つ高度にコスト有効に組込まれ、これらのデッドロックの発生を伝送帯域幅の僅かな低減により、防止することが理解されよう。従って、本発明は実質的に任意のサイズのパケット・ネットワークにおいて即刻使用され、公衆または専用電話回線（例えば局所、広域または首都圏ネットワーク）または他の類似のネットワークなどのデジタル通信、或いは大容量並列処理システムの通信中枢などの特殊アプリケーションに関わり無く、広範且つ様々な範囲のパケット交換環境に渡って使用される。しかしながら、後述の説明を単純化するために、本発明は大容量並列処理システム、そして特に、ＩＢＭにより今日製造されるスケーラブル並列処理システムのＳＰファミリにおいて使用されるIBM9076 SP-1高性能通信ネットワークにおいて使用されるように述べられる。

【００２４】本発明の理解を容易にするために、最初に並列処理システムのパケット経路指定の様々な態様、特に、そこで使用される双方向クロスポイント方式パケット・ネットワークに関する態様について述べ、次に典型的な経路指定デッドロック状況について、そして最後に、これらのデッドロックの発生を有利に防止する本発明について詳細に述べることにする。

【００２５】最初に、図１に示される従来の３２プロセッサ並列処理システム５について考えてみる。このシステムは３２個のノード・パケット・スイッチ１００（ここでは”パケット・ネットワーク”または単に”ネットワーク”としても参照される）を含み、各ノードには３２個の別々の（しかしながら一般には同一の）処理要素１１０（特に処理要素１１０_{0}、１１０_{1}、．．．、１１０_{31}）が接続される。各要素はシステムの処理ノードを形成する。ネットワークはこれらの処理ノードの１つから他のノードへの高速伝送を提供する。処理要素自身はそれぞれマイクロプロセッサを基本とし、通常、ＩＢＭにより製造されるRS6000 RISCマイクロプロセッサを使用する。本発明は任意のこれらの要素のアーキテクチャまたは回路には無関係であるので、当業者には容易に明らかとなろうこれらの態様については詳細には述べない。しかしながら、本発明は後に詳述されるように、これらの処理要素の１つにおいて実行されるシステム初期化ソフトウェア、及びこれらの各々の要素内に記憶される経路テーブル内において実現される。従って、これらの特定の態様については、特に後述される。

【００２６】図示のように、ネットワーク１００は８個の別々の８×８双方向スイッチ回路１２０により構成され、これらは２つの相互接続ステージ、すなわち４個のスイッチ回路１２０_{0}、１２０_{1}、１２０_{2}及び１２０_{3}を含む”入力”ステージと、４個のスイッチ回路１２０_{4}、１２０_{5}、１２０_{6}及び１２０_{7}を含む”出力”ステージとに編成される。”入力”及び”出力”の指定は、純粋に説明の都合において任意であり、実際には、ネットワーク上のステージまたはポートは入力または出力ステージ或いはポートとして機能する。これらの各々のスイッチ回路

は、好適には、中央キュー・ベースの非ブロッキング８ウェイ・ルータである。各スイッチ回路は単一の集積回路として、すなわち、いわゆる"チップ"として集積化され、ここでは各こうしたスイッチ回路自身を"スイッチ・チップ"として参照する。もちろん当業者には理解されるように、各スイッチ回路は単一のチップとしてだけ具体化される必要はない。いずれの場合にも、スイッチ・チップ自身は本発明の１部を形成しないので、これについては詳細には述べないことにして、この回路のその他の詳細に関して述べることとする。図示のように、各スイッチ・チップは中央キューを含み、これらは対応するスイッチ回路１２０$_0$、１２０$_1$、１２０$_2$、．．．、１２０$_7$内のキュー１３０$_0$、１３０$_1$、１３０$_2$、．．．、１３０$_7$として表される。基本的に各中央キューの目的は、とりわけ入力阻止及びデッドロックを改良するために、対応するスイッチ回路を通過する別の経路を提供することであり、後者すなわちデッドロックは、入力ポート（特に内部のＦＩＦＯバッファ）及び逆のトラフィックにより充填されたキューに起因する（これは本発明が対象とするのとは異なる形態のデッドロックである）。

【００２７】ネットワークの入力及び出力ステージは、接続マトリックス１４０を介して相互接続され、これらの各々の接続は実質的にバイト幅物理リンク（ケーブル）であり、特にそれらの内のリンク１４０$_0$、１４０$_1$、１４０$_2$及び１４０$_3$が番号付けされて示される。このマトリックスを介して、入力ステージ内のそれぞれのスイッチ・チップのポートが別々にまた物理的に出力ステージ内のあらゆるスイッチ・チップの対応するポートに接続される。例えば、スイッチ・チップ１２０$_0$はポート０乃至７を備え、そのポート４乃至７を通じ、対応するケーブルを介して、各スイッチ・チップ１２０$_4$、１２０$_5$、１２０$_6$及び１２０$_7$上のポート４に接続される。８個のスイッチ・チップ及び接続マトリックス１４０を含むパケット・スイッチ１００は、集合的に単一のスイッチ・ボードを含む。各スイッチ・チップのポート０乃至３はスイッチ・ボード外のリンクに接続され、各スイッチ・チップのポート４乃至７は、接続マトリックス１４０内のリンク（ケーブル）に接続され、それを介して同一ボード内の別のスイッチ・チップのポートに接続される。

【００２８】ある要素が別の要素からデータを要求したり、データを供給したりするなど処理要素が互いに通信するために、"出所"処理要素は自身が実行するアプリケーション部分にもとづき、命令またはデータと共に適切なメッセージを含むパケットを形成し、そのパケットを"宛先"処理要素に伝送するためにパケット・スイッチ１００に送信する。宛先要素はパケット内に含まれるデータまたは命令を処理し、適切な応答を生成する。応答は次に、宛先処理要素において実行されるアプリケーション部分にもとづき、別のパケットに形成され、例えば出所または異なる処理要素に伝送して処理するためにネットワークに返送される。

【００２９】ネットワークを介するパケット伝送を容易にするために、各パケットは経路バイト形式の特定の経路指定命令を有するヘッダを含む。後述のように、全ての経路が予め定義される。出所処理要素がアセンブル中の任意のパケットの宛先を決定すると、その要素は単に、宛先処理要素をアドレスとして有するその内部（局所）経路テーブルをアクセスし、適切な経路バイト値の形式で経路を読出す。この値が単に経路バイトとしてパケットのヘッダに挿入される。

【００３０】図２はパケット・ネットワークを通じて伝送される典型的なパケット、すなわちパケット２００の構成を示す。個々のパケットは例えば２５５バイト長である。図示のように、パケット２００は連続するフィールド、すなわち長さフィールド２１０、経路フィールド２２０（それ自身経路バイト２２０$_1$、２２０$_2$、．．．、２２０$_n$を含む）、シーケンス番号フィールド２３０及びデータ・フィールド２４０を含む。長さフィールド２１０は、パケット長をバイトで指定する８ビット・ボリュームを含む。経路フィールド２２０は複数のバイト、特に経路バイト２２０$_1$、２２０$_2$、．．．、２２０$_n$を含み、これらは集合的にパケットがネットワーク全体を通じてその出所ノードから宛先ノードに至る特定の単一の経路（パス）を指定する。フィールド２３０は出所処理要素により提供されるシーケンス番号を保持する。この番号は、このパケットに対応して出所処理要素により割当てられ、宛先処理要素により使用され、所与のシーケンスにおけるパケットの順番を識別する。この番号自体は、宛先におけるシーケンス外のパケットの処理の防止のためのチェックに使用される。データ・フィールド２４０は連続するバイトを含み、これらは集合的にパケットにより宛先処理ノードに伝搬されるデータ（実際のデータまたは命令を含む）を形成する。フィールド２１０、２２０及び２３０は集合的にパケット・ヘッダを形成する。

【００３１】経路指定フィールド２２０に現れる経路バイトの数（ｎ）は、パケットが通過する交換ステージの数により決定される。その点に関し、各経路バイトは２つの連続スイッチ・チップに対応する経路指定命令を保持する。従って、パケットが宛先処理ノードに達するまでに、図１に示されるように、ネットワーク内の２つの連続ステージ内の２個のスイッチ・チップを通過するだけであれば、フィールド２２０は経路バイト２２０$_1$だけを含むことになる。レイヤ・ネットワーク（layer network）においては、追加の対のスイッチ・チップが使用される。全ての経路バイトが同一の形式を有する。この点に関し、経路バイト（Ｒ［７：０］）は１ビットのフィールド選択子（Ｒ［７］、図示せず）、及び２つの

３ビットの経路フィールド（Ｒ［６：４］及びＲ［２：０］、両者共に図示せず）を含む。ビットＲ［７］の値が０の場合、スイッチ・チップはパケットを２進値Ｒ［６：４］により指定されるそのチップ上の出力ポートに経路指定し、次にビットＲ［７］の値を１に設定する。或いはビットＲ［７］の値が１の場合、スイッチ・チップはパケットをビットＲ［２：０］で指定されるそのチップ上の出力ポートに経路指定し、その間に、この完全な経路バイトを廃棄する。このようにして、パケットから経路バイトを解析する。従って、各経路バイトは２つの連続スイッチ・チップに対する経路指定命令を提供する。ｎ個の経路バイトを経路フィールド２２０内に連結することにより、各パケットはスイッチ・チップの最大２ｎのステージを通じて経路指定される。

【００３２】要約すると、パケットを受信するスイッチ・チップは、そのパケット内にその時、存在する第１の経路バイトを調査し、そのパケットをそのバイトにより示されるポートに経路指定する。そうする間に、そのパケットのパス内のあらゆる別のスイッチ・チップは、その完全な経路バイトをパケットから切り取る（除去する）。これは次に、経路フィールド２２０内の次に続く経路バイトを、次のスイッチ・チップ及び交換ステージに対応する第１の経路バイトとして形成する。宛先処理ノードに到来する時、パケットは経路バイトを含んでいない。各スイッチ・チップはその時パケットにより伝搬される第１バイト以後の追加の経路バイトを意識せず、第１バイトに対してその回路は、その特定の経路指定を実行する。更に各スイッチ・チップは第１バイト以外の経路バイトと、続くデータ・バイトとを区別しない。

【００３３】上述のように、経路指定はパケット・アセンブリの間に最初に予め定義された経路バイトをパケット・ヘッダに挿入し、次にそのパケットの実際の経路指定が導かれ指令されることにより、出所処理要素及び宛先処理要素に関係なく、これらの各々のバイトの特定の値によりネットワーク内において達成される。

【００３４】図３は、図１に示されるシステム５を構成する処理ノード１１０を示し、特に、これらのノードのメモリ内に存在してパケット経路指定を実行する様々なファイル及びテーブルを示す。パケット・スイッチ（ネットワーク）１００は時分割の２つのモードで機能する。それらの一方は実行フェーズであり、この間、スイッチ回路は単に入来パケットを経路指定する。他はサービス・フェーズであり、この間、プロセッサは初期化されるか、ネットワークが回線交換方式でモニタ及び管理される。ネットワークに接続される全てのスイッチが、モード間で同期化ロック・ステップ方式で転送する。実行フェーズの間、特定の処理要素は特定のタスクを任せられる。例えば、処理要素１１０_{0}及び１１０_{1}は、システム５から他のネットワークへの、または処理システムへのリンクを提供し、それらの間で情報を転送するための入出力ノードとして指定される。他の処理要素、例えば処理要素１１０_{2}、１１０_{3}、．．．、１１０_{31}は、全て実際のアプリケーション処理のための計算ノードとして使用される。処理要素の１つ、例えば処理要素１１０_{31}は、サービス・フェーズの間の様々なネットワークオペレーションを引受けるサービス・プロセッサとして使用される。必要に応じて実行フェーズの間、サービス・プロセッサは計算ノードとしても機能することができる。サービス・プロセッサはハードウェア的見地からは、他の全ての処理要素と同一であるが、サービス・プロセッサはそのメモリ（ここではメモリ３４０）内に、サービス・フェーズの間に実行される追加のソフトウェア、とりわけ初期化ルーチン３７０を含み、これを実行する。例えばこのフェーズは全てのスイッチ回路及びネットワークに接続される全ての他の装置（全ての他の処理要素を含む）に対して、初期化、通信リンク同期、大域時間同期、故障判断、及び分離、及び様々な診断サービスを提供する。初期化機能はサービス・フェーズの１部に過ぎないのでサービス・フェーズのこの部分、特にパケット経路指定及び本発明に関連する観点についてのみ、以降で述べることにする。初期化フェーズは、システムが任意のアプリケーション処理を請け負う以前に、請け負わされる。

【００３５】サービス・プロセッサ１１０_{31}はそのメモリ３４０内に、ネットワークに接続される全ての処理要素を含むそれぞれの及びあらゆる装置、及びこれらの装置をリンクするためにネットワーク内において使用される特定の双方向物理接続（ケーブル）を集合的に定義する構造化エントリのデータベース、特にトポロジ・ファイル３５０を記憶する。データベースが生成される方法は本発明には関連しないので、ここでは触れないことにする。トポロジ・ファイルにおいて、スイッチ回路及び他の装置の最大数が装置エントリにより最初に識別され、任意のこれらの回路及び装置間に存在する各物理接続のエントリがそれに続く。装置エントリは２つの数値フィールドを含み、これらは”装置番号（n_v）；スイッチ回路番号（n_s）”の形式を取る。これらの値が提供されると装置識別（ｉｄ）の番号付けが０乃至n_vの範囲において、またスイッチ回路ｉｄの番号付けが０乃至n_sの範囲において仮定される。最大１６個の装置及び８個のスイッチ回路を含むネットワークでは、装置エントリは単に”16 8”である。各接続エントリは６つのフィールドを有し、これは”装置１タイプ；装置１ｉｄ；装置１ポート；装置２タイプ；装置２ｉｄ；装置２ポート”の形式を取る。装置タイプ情報は装置の性質、すなわちその装置が処理要素かどうかを指定し、そうであれば、その要素がサービス・プロセッサかどうか、或いはスイッチ回路かどうかを指定する。接続エントリの例は”tb0 14 0 s 3 6”であり、これは”ｉｄ１４の処理要素が全２重方式で、そのポート０からスイッチ回路３の入出力

両ポート６に接続される”ことを意味する。ネットワークの配線は、通常、極めて規則的であり、良好に定義され対称的である。しかしながら、実際には幾つかのスイッチ・ボードは、保守状態或いは故障状態の結果として、故意に分離される他のネットワーク・コンポーネント、例えばケーブル、スイッチ回路（特に使用されるスイッチ・チップ）または処理要素のために、パワー・ダウン状態の可能性がある。従って、任意の瞬間におけるネットワーク・トポロジは極めて不規則であったりする。

【００３６】いずれにしても、初期化及び特に初期化ルーチン３７０の実行の間、サービス・プロセッサ１１０_{31}はその時存在するトポロジ・ファイル３５０を読出し、次にテスト・メッセージを同報し、それに対応する応答を受信することにより、ネットワークに接続される各装置と同様、ネットワーク内の各接続の状態を物理的に判断する。これらの応答にもとづき、サービス・プロセッサは、例えば既知のブレッドス・ファースト探索（breadth-first search）により、ネットワークの各（出所）ノードをネットワークのあらゆる他の（宛先）ノードに接続するための全ての使用可能な経路を判断する。双方向マルチステージ・クロスポイントネットワークに固有のパス冗長性により、異なるスイッチ・ステージ内の異なるスイッチ回路を通過して、１対の出所ノード及び宛先ノードを接続する複数の経路がしばしば存在する。各共通の出所／宛先ノード対間の複数の経路を鑑み、サービス・プロセッサは次にこれらの各々のノード対に対応するこれらの経路の１つを選択し、その経路をメモリ３４０内の大域経路テーブル３６０に記憶する。これらの経路はネットワーク内におけるトラフィック渋滞及びホット・スポットを回避するために単位時間に渡り、ネットワーク全体を通じてパケット・トラフィックの実質的に一様な分布を達成するように、主に最短パスにもとづき選択される。

【００３７】ネットワーク１００の使用可能な各出所／宛先ノード対間のパスを定義する大域経路テーブル３６０が完全に構成されると、サービス・プロセッサ１１０_{31}は次にネットワークを通じ、そのテーブルの対応部分を自身を含む各個々の処理要素に局所経路テーブルとして、そこに記憶するために提供する。この部分は、その特定の処理要素を出所ノードとしてリストする経路だけを含む。従って、例えば処理要素１１０_{0}はそのメモリ３１０内に、局所経路テーブル３２０を記憶し、サービス・プロセッサ１１０_{31}はそのメモリ３４０内に、局所経路テーブル３８０を記憶する。他の処理要素についても同様である。パケット形成の間、上述のように、各処理要素は単にその局所経路テーブルをアクセスするだけで、その時アセンブルされるパケットの宛先にもとづき、その宛先の経路指定バイトの値をテーブルからそのパケットのヘッダにコピーする。

【００３８】上述の説明を考慮して、経路指定デッドロックを表す図１を再度参照することにする。

【００３９】経路指定デッドロックは、各々が例えばスイッチ・チップの異なる交換ステージ内の中央キューに存在する対応パケットが、連続するステージ内のスイッチ・チップ対を接続する共通パス上における経路指定を同時に待機する度に発生する。従って、ここで”A”と記されるパケットがスイッチ・チップ１２０_{0}の中央キュー１３０_{0}に内在し、処理ノード１１０_{0}から”丸A”で示される破線のパスを介して、処理ノード１１０_{4}に経路指定されるのを待機しているものと仮定する。このパスを通じ、パケット”A”はスイッチ・チップ１２０_{0}により、ケーブル１４０_{0}を介してスイッチ・チップ１２０_{4}のポート４に導かれ、次にこの後者のチップのポート５及びケーブル１４０_{1}を介して、入力ステージ特に処理ノード１１０_{4}に接続されるスイッチ・チップ１２０_{1}のポート０に経路指定されて戻される。同様にキュー１３０_{0}に内在するパケット”A”と同時に、スイッチ・チップ１２０_{4}、１２０_{1}及び１２０_{5}のそれぞれの中央キュー１３０_{4}、１３０_{1}及び１３０_{5}に、３つの他のパケット”B”、”C”及び”D”が内在するものと仮定する。パケット”B”はスイッチ・チップ１２０_{4}のノード１に接続される処理要素１１０_{17}から、”丸B”で示される破線のパスを介して、スイッチ・チップ１２０_{5}のノード３に接続される処理要素１１０_{21}に経路指定される。同様にパケット”C”は、スイッチ・チップ１２０_{1}のノード２に接続される処理要素１１０_{6}から、”丸C”で示される破線のパスを介して、スイッチ・チップ１２０_{0}のノード２に接続される処理要素１１０_{2}に経路指定される。同様にパケット”D”は、スイッチ・チップ１２０_{5}のノード１に接続される処理要素１１０_{21}から、”丸D”で示される破線のパスを介して、スイッチ・チップ１２０_{4}のノード０に接続される処理要素１１０_{16}に経路指定される。

【００４０】図示のように、全ての４つのパケットは同時に衝突する経路を有し、同一セットの４つのケーブルを介する。各経路はそのケーブルを他の２つの経路と共用する。結果的に、各スイッチ・チップ１２０_{0}、１２０_{1}、１２０_{4}及び１２０_{5}は、対応する中央キューに内在するこれらのパケットと共に、これらのスイッチ・チップの任意の他の１つが最初にそのパケットを経路指定するのを待機することになる。各パケットは基本的にスイッチ・チップの１つにおいて（但し、異なるポートを通じて）その方向を反転する、すなわち”ターン・アラウンド”するので、これらの全てのパケットにより取られる経路は、集合的に閉ループ・パターン（番号Ｉ－ＩＩ－ＩＩＩ－ＩＶで示され、ここでは”サイクル”として参照される）を形成することになる。スイッチ・チップはこれらのどの特定のパケットを最初に経路指定するかを決定できないので、全てのスイッチ・チップは単に待

機し、いずれのパケットも経路指定されない。サイクル内の４つの各々のパケット自身が、残りの３つのパケットを妨害することになる。結果的に、経路指定デッドロックが発生する。このデッドロックが持続する間、対応するパスはパケット・トラフィックを伝搬しない。従って、処理要素110_4、110_{23}、110_2及び110_{16}は単にパケットの到来を待機し、これらのパケットを要求するアプリケーション部分の処理が延期される。これはすなわち、システム５の処理スループットを低下させることになる。経路指定デッドロックが発生すると、この状態は何らかの手段により解決されるまで無期限に継続する。経路指定デッドロックは比較的稀にしか発生しないが、並列処理システムの規模が増大すると、これらのデッドロックの発生の潜在性も増加する。

【００４１】この現象を鑑み、本発明は比較的大規模な大容量並列処理システムにおいて、経路指定デッドロックの発生を防止する手法を提供する。本手法は具体化が非常に単純で高度にコスト有効であり、パケット・ネットワークにおける伝送帯域幅の適度で受諾可能な低減を強要するに過ぎない。

【００４２】本手法により、特定の予め定義された経路が、大域経路テーブルの形成の間にこれらの経路を使用する特定の処理要素（ネットワーク・ノード）の相対ロケーションにもとづき考慮から除外される。禁止された経路は、もしそれらが使用されないと、閉ループ経路指定パターン、従って経路指定デッドロックの発生を防止する経路として選択される。経路テーブルに含まれる経路の選択において、システムのある区分、例えばシステムの半分内のノード間をもっぱら流れるパケット・トラフィックを別の区分、例えばシステムの他の半分内のノード間を流れるパケット・トラフィックから分離するために経路が禁止される。この点に関し、システムの共通区分内のノード間を通過するパケットの経路を抽出するために、システムの別の区分を通過するパス（ケーブルなど）を含む経路が禁止される。複数のシステム区分、例えばシステムの異なる半分内のノード間でパケットを伝搬する経路の選択においては、こうした経路の禁止は発生しない。システムの各区分、例えば半分を他の区分、例えばシステムの他の半分内の処理要素対間をもっぱら流れるパケット・トラフィックから分離することにより、これらのパケットの相互作用により生じる経路指定デッドロックが有利に防止される。

【００４３】比較的大規模な大容量並列処理システム、例えば５１２の別々の処理要素を使用するシステムにおいて必要なプロセッサ間経路指定機能を提供するために、システムは多数のスイッチ・ボードを使用する。各スイッチ・ボードは上述されたように同一であり、２つの相互接続ステージ、すなわち個々の処理要素に接続されるノード・スイッチ・ボード（ＮＳＢ）、及びノード・スイッチ・ボード自身を相互接続するために使用される中間スイッチ・ボード（ＩＳＢ）に編成される。５１２プロセッサ・システムは、通常、４８個の別々のスイッチ・ボードを使用し、これらの内の３２個のボードはＮＳＢ専用であり、残りの１６個のボードはＩＳＢ専用である。各ＮＳＢは１６個のそれぞれ異なる処理要素に接続される１６ポートと、１６個の各ＩＳＢ上の異なるポートに相互接続される別の１６ポートを提供する。この構成では、ＮＳＢはパケットを自身が接続される個々の処理要素との間で経路指定し、ＩＳＢはパケットを異なるＮＳＢ間で経路指定し、全ての完全な経路が、上述のようにパケット・ヘッダに含まれる経路指定バイトにより指定される。

【００４４】５１２プロセッサ・システムの例が、図４にシステム４００として示される。図示のように、このシステムは集合的に処理ノード４１０として示される５１２の異なる処理要素415_0、．．．、415_{15}、．．．415_{496}、．．．415_{511}を提供し、物理的見地から１６個の処理要素を含む３２個の物理ラック、特に処理ラック410_0、．．．410_{31}に編成される。各ラックはそれぞれのＮＳＢの１６ポートに接続される。システム４００は３２個の$ＮＳＢ440_0$、440_1、440_2、440_3、440_4、440_5、．．．、440_{30}及び440_{31}（ＮＳＢ０、ＮＳＢ１などとしても指定される）を含む。各ＮＳＢの残りの１６ポートは、接続マトリックス４５０内の個々のケーブルを介して、１６個のＩＳＢ４６０、特に$ＩＳＢ460_0$、460_1、460_2、．．．、460_{15}（ＩＳＢ０、ＩＳＢ１などとしても指定される）の各々の対応するポートに相互接続される。例えば、$ＮＳＢ440_0$（ＮＳＢ０）上の１６個の各ポートは、１６個のＩＳＢの対応する異なる１つのポート０に接続されるように示され、それにより$ＮＳＢ440_0$は各ＩＳＢにパケットを経路指定できる。他のＮＳＢについても図示のように、あらゆるＩＳＢに同様に相互接続される。ＩＳＢであろうとＮＳＢであろうと、全てのスイッチ・ボードは互いに同一であるが、接続マトリックス４５０を明瞭に表す都合上、ＩＳＢはＮＳＢと異なるように示される。

【００４５】システム４００において、中間スイッチ・ボードなどの使用に頼る他の大規模大容量並列処理システムと同様、本発明を使用しないと、経路指定デッドロックが発生することが理解される。なぜなら、パケットが異なるＮＳＢ間で経路指定される時、図１に示されるシステム５のスイッチ120_4及び120_5内で、パケット”A”及び”C”がそれらの方向を反転（”ターン・アラウンド”）する時のように、その方向をＩＳＢ内で反転するからである。図４に示されるように、パケットはＩＳＢ内においては生成されず、単にそれを通じて別々のＮＳＢ間で経路指定されるだけなので、閉ループ経路指定パターンがもしも発生すると、それらはＩＳＢに延びる必要があり、ＮＳＢ内だけに存在するとは限らなくな

る。システム４００内の経路指定デッドロックは、任意の１つまたは複数のＮＳＢ自身だけに制約されない。

【００４６】本発明の教示によれば、禁止する経路を決定するために、システム４００は例証的に半分に区分される。この場合、１６個の連続するＮＳＢ（例えばＮＳＢ０乃至１５、及びＮＳＢ１６乃至３１）、及び２５６個の連続する処理要素（例えばそれぞれ要素４１５$_0$、．．．、４１５$_{255}$及び４１５$_{256}$、．．．、４１５$_{511}$）は各半分に割当てられる。また最初の８個のＩＳＢが片方の半分に含まれ、残りの８個のＩＳＢが他の半分に含まれる。この点に関し、図５を参照すると、図４のシステム４００を構成する全てのＮＳＢ並びにＩＳＢ４６０が示される。図示のように、システムは５０３と５０７のそれぞれ半分に区分される。３２個の各ＮＳＢ上のポート０などの共通ポート（ラベル付けされていない）が、別々の対応するパス（ケーブル）を介して、単一のＩＳＢ上の３２個のポートの対応する１つに接続される。全てのＮＳＢ上の残りの各ポート及び他のＩＳＢについても同様である。システムの半分５０３は、ＮＳＢ４４０$_0$乃至４４０$_{15}$及びＩＳＢ４６０$_0$乃至４６０$_{15}$を含む。ここではＮＳＢ４４０$_0$乃至４４０$_{15}$は、パス５１０$_{0,0}$、５１０$_{1,0}$、．．．、５１０$_{15,0}$を介して、単一のＩＳＢの１６個の連続するポート（特にそれらの３つだけが示されている）、ここではＩＳＢ４６０$_0$の特にスイッチ・チップ５３０$_0$乃至５３０$_3$に接続されるように示される。残りのシステムの半分５０７は、ＮＳＢ４４０$_{17}$乃至４４０$_{31}$、及びＩＳＢ４６０$_8$乃至４６０$_{15}$を含む。同様にこれらの特定のＮＳＢはパス５１０$_{16,15}$、５１０$_{17,15}$、．．．、５１０$_{31,15}$を介して、単一のＩＳＢの対応するポート、ここではＩＳＢ４６０$_{15}$の特にスイッチ・チップ５４０$_4$乃至５４０$_7$に接続されるように示される。

【００４７】システムの共通半分、例えば半分５０３内に配置される処理要素間を通過するパケットに対しては、システムのその半分内に完全に含まれるＩＳＢを含む使用可能な経路（ＮＳＢ４４０$_0$と４４０$_{15}$間の経路５２２、及びＮＳＢ４４０$_{31}$と４４０$_{16}$間の経路５２４など）だけが許可され、他の経路（ＮＳＢ４４０$_0$と４４０$_{15}$間の破線で示される経路５３４、及びＮＳＢ４４０$_{31}$と４４０$_{16}$間の破線で示される経路５３２）は禁止される。従って、後者の任意の経路はシステム初期化の間に、これらの処理要素を接続する大域経路テーブル内に含まれない。禁止された経路はまた、その経路上の”Ｘ”により示される。或いはシステムの異なる半分内に配置される処理要素間を通過するパケットに対して、こうした経路が禁止されなくてもよい。この場合、経路選択は大域経路テーブル内に結果的に含むものに対して、システムの半分にもとづく制限無しに、ＮＳＢ４４０$_0$と４４０$_{16}$間のその時使用可能な全ての経路（１つの場合もある（特に図示せず））の中から実施される。

【００４８】経路の禁止は後述されるように、大域経路テーブルが生成される間に、特定の経路指定指示を処理することにより実行される。この処理は、全ての禁止経路がネットワーク・ノードの所与の対間で定義される経路として選択されることを防止する。

【００４９】自身の内部処理要素間で発生するパケット・トラフィックに対応して、システムの各半分を分離し、それにより他の半分に含まれる処理要素間を通過するパケットとの相互作用を排除することにより、経路指定デッドロックが有利に防止される。

【００５０】驚くことに、上述のように５１２ポート交換ネットワークの多数の分析の結果、ネットワークを通過する期待されるトラフィック・パターンに関し、本発明の手法によれば、ネットワークの最大伝送帯域幅が支障無い程度に減少するだけであることが判明した。この点に関し、本発明の使用は、ネットワークにおいて使用可能な最大帯域幅のほぼ７４％を確保し、これは期待した約５０％を大きく上回るものである。従って、経路指定デッドロックを回避するための本手法の使用による不利益は、特に獲得される利点を鑑みれば、極めて受諾可能と言える。

【００５１】上述の説明を鑑み、図６は、図４に示されるシステム４００内に配置されるサービス・プロセッサ（例として処理要素４１５$_{511}$）内で実行される、本発明の教示によりパケット経路を定義する経路テーブル発生器ルーチン６００のハイレベル流れ図を示す。ルーチン６００は、上述のようにサービス・プロセッサ内で実行される初期化ルーチンの１部である。

【００５２】図６に示されるように、ルーチン６００へのエントリに際し、実行は最初にブロック６１０に移行し、トポロジ・ファイル及び付随する経路指定指示を読出す。デッドロック回避経路指定を提供するために、パケット・ネットワーク内の各装置、例えばスイッチ回路（または特にそこで使用されるスイッチ・チップ）に対応する適切な指示、すなわちその装置を通過する経路指定が制限されているか否か、換言すると、パケットがこの回路を通じて方向を反転可能か否かを示す指示が、トポロジ・ファイル内に含まれなければならない。ネットワーク１００を実現する図１に示されるスイッチ・ボードについて考えてみる。上述のように、各スイッチ・チップのポート０乃至３は、スイッチ・ボードの外部のリンクに接続され、各スイッチ・チップのポート４乃至７は接続マトリックス１４０内のリンク（ケーブル）に接続され、それを通じて、同一ボード内の別のスイッチ・チップのポートに接続される。トポロジ・ファイル内において特定のスイッチ・チップに対して指定される経路指定指示”ｎｒ”は、そのチップに関し、経路指定制限が存在しないことを意味する。パケットはこのチップ上の任意の８個のポートに入力することができ、チップ上の他のポートから去ることができる。この例では、パケッ

トはチップ内でその方向を反転（”ターン・アラウンド”）することができる。或いはスイッチ・チップに対して、トポロジ・ファイル内に経路指定指示”n-i-t”が存在すると、ポート４乃至７に入力するパケットは、チップ上のポート０乃至３からだけ出力するようにその経路指定が制限される。すなわち、そのパケットはチップ内で方向を反転することを禁止される。しかしながら、”n-i-t”指示は、スイッチ・チップの任意のポート０乃至３に到来するパケットに対しては制限せず、これはそのチップ上の任意の他のポートに経路指定される。デッドロック回避指示を有するトポロジ・ファイル内のサンプル行は次の通りである。

【数１】aux routing n-i-t 330 331 332 333

ここで、”aux routing”は、経路指定指示を有する補助行を意味する。そして、”330 331 332 333”は、トポロジ・ファイル内で使用されるフォーマットの特定のスイッチ回路の数値識別子である。

【００５３】ブロック６１０が完全に実行されると、実行はブロック６２０に移行し、トポロジ・ファイル内で指定される各ケーブル（リンク）に関連する重みを０に設定する。更に出所ノード・カウンタに相当するノードｉが０に初期化される。その後、実行はブロック６３０に移行する。この特定のブロックはトポロジ・ファイル内に含まれるデータと付随するデッドロック回避経路指定指示とを一緒に使用することにより、パケット・ネットワークを通じて現出所ノード（ノードｉ）をシステム内のあらゆる宛先ノードに接続するために使用可能な経路のセットを抽出する。具体的には、既知のブレッドス・ファースト探索（breadth-first search）により、最短長を有する経路、すなわち必ずしも物理的に最短長を有するわけではないが、最少の個々のリンク（ケーブル）を有する経路が選択される。上述された各スイッチ回路に関連するデッドロック回避経路選択を表す擬似コードを次に示す。

【数２】case of routing_directive is
```
{
”nr”:total_permissible_oports=8;
/*スイッチ・チップ上の全出力ポート*/
”n-i-t”:if(input_port<4)
total_permissible_oports=8;
else
total_permissible_oports=4;
}
i=0;
while(i<total_permissible_oports) do
{
permissible_oport[i]=i
i=i+1
}
```

【００５４】最短パス経路のセットの選択は、ブロック６４０に示されるように、経路が現出所ノードから全ての宛先ノードに延びるまで、すなわち経路が宛先ベースになるまで出所ベースで発生する。１つの最短長経路だけが出所ノードから宛先ノードに生じる場合、その経路が選択されて使用される。或いは複数のこうした経路がこの出所ノードと共通宛先ノード間で生じる場合、集合的に最低の重みのケーブルを有する経路が選択される。重みベースの選択により、パケット・ネットワーク全体を通じて最小のケーブルの共用を維持するように、トラフィック負荷が平衡される。出所ノードと宛先ノードの間で特性の経路が選択されると、その経路内の各ケーブルに関連する重みが１だけ増分される。ブロック６３０及び６４０は、理解を容易にするために別個のブロックとして示されるが、オペレーションは一般に結合される。

【００５５】全ての宛先ノードに対応して全ての経路が選択されると、実行はブロック６５０に移行し、全ての選択経路を大域経路テーブルに書込む。これにより現出所ノードに対する経路テーブルが形成される。その後、実行は判断ブロック６６０に移行し、ネットワーク内のあらゆるノードに対して、経路テーブルが大域経路テーブルに書込まれたかどうかを判断する。経路テーブルがあらゆるノードに対して書込まれていない場合には、判断ブロック６６０は実行を否定パス６６７を介して、ブロック６７０に移行させる。この後者のブロックの実行により、出所ノード・カウンタｉが１増分される。実行は次にパス６７５を介して、ブロック６３０へループして戻り、次に続くノードの経路を判断しそれを書込む。或いは経路テーブルが全てのノードについて書込まれると、実行は判断ブロック６６０からの肯定パス６６３を介して、ルーチン６００を終了する。このルーチンの実行の後、初期化処理の完了に先立ち、上述のサービス・プロセッサはネットワークを通じ、大域経路テーブルの対応部分を、自身を含む各々の及びあらゆる個々の処理要素に提供する（特にコピーする）。そして、こうして記憶されたものが、後に局所経路テーブルとして使用される。この部分は、その特定の処理要素が出所ノードの時に選択される経路を含むだけである。

【００５６】これまでの説明から当業者には理解されるように、本発明は５１２の別々の処理要素を有する大容量並列処理システムに関連して述べられてきたが、もちろんこれに限るものではない。実際に、本発明は実質的に双方向マルチステージ相互接続クロスポイント・ベース・パケット・ネットワークを使用する任意のサイズの並列処理システムにおける、経路指定デッドロックの回避にも適用される。その点に関し、本発明は６４プロセッサ・システム、２５６プロセッサ・システム、及び他のサイズの類似のシステム、並びにマルチステージ相互接続クロスポイント・パケット・ネットワークを使用する他のシステムにそれらの最終利用に関係無く、容易に

組込むことが可能である。

【００５７】更に、本発明の教示はパケット・ネットワークを２つの別々の半分に区分し、それらの間の経路指定を制限する状況において述べられたが、こうしたネットワークは本発明により任意の数の別々の区分に分割され、これらの各区分内だけをもっぱら通過するパケット・トラフィックを分離するように編成される。もちろん、区分数が増加すると、それに伴い区分化を達成するために必要となる禁止経路の数も増加する。残念ながら、禁止経路の数が増えるとパケット・トラフィックを伝搬する使用可能な経路が減少し、従って、ネットワークの伝送帯域幅が減少する。支障の無い帯域幅の減少を鑑みると、達成されるパケット分離及びデッドロック回避の点から、２つの区分が優れたトレードオフを提供することが判明した。

【００５８】まとめとして、本発明の構成に関して以下の事項を開示する。

【００５９】（１）パケット・ネットワークの外部の複数のノードを集合的に相互接続するクロスポイント・スイッチの連続ステージを含む前記ネットワークを有する装置において、パケットが前記ネットワーク及び少なくとも１つの前記スイッチを介して、規定経路上を第１の前記ノードから第２の前記ノードに伝搬されるものにおいて、前記ネットワーク内における経路指定デッドロックの発生を回避する実質的な方法であって、パケットが前記複数のノード内の個々のノードから、異なる対応する前記経路を介して、前記複数のノードのあらゆる他のノードに伝搬されるように、前記ネットワークを介する複数の規定経路を第１に定義するステップであって、前記の各定義経路が少なくとも１つのリンクに伸び、第１のネットワーク区分だけに接続される第１及び第２の前記ノード間を通過するパケットが、第２のネットワーク区分に伸びるリンクを有する経路上で伝搬されないように、前記ネットワークを前記第１及び前記第２のネットワーク区分に分割するように前記規定経路を定義する、前記第１の定義ステップと、全ての前記規定経路を結果の経路テーブルに記憶するステップとを含む、方法。

（２）前記第１の定義ステップが、前記第１及び前記第２のネットワーク区分にそれぞれ接続される第３及び第４のノード間を通過するパケットが、前記第１及び前記第２のネットワーク区分間に伸びる少なくとも１つのリンクを有する経路上で伝搬されるように、前記複数の規定経路を定義する第２の定義ステップを含む、前記（１）記載の方法。

（３）前記ネットワーク内において出所ノードから宛先ノードに経路指定されるパケットをアセンブルする際に、前記第３及び前記第４の両ノード内において、前記パケットの結果的経路を生成するために、前記経路テーブルをアクセスするステップと、前記結果的経路を前記パケットにコピーするステップと、前記パケットを前記結果的経路上で前記ネットワークを介して経路指定するステップとを含む、前記（２）記載の方法。

（４）前記経路テーブルの異なる部分を、前記複数の各ノードに対応する別々の局所経路テーブルにダウンロードするステップであって、前記の各経路テーブル部分が前記各ノードを出所ノードとして有する全ての前記規定経路を指定する、前記ダウンロード・ステップを含み、前記経路コピー・ステップが、前記出所ノードから前記宛先ノードに伝搬されるパケットの前記結果的経路を生成するために、前記パケットの前記宛先ノードにもとづき、前記出所ノードの前記局所経路テーブルをアクセスするステップを含む、前記（３）記載の方法。

（５）前記の各パケットが少なくとも１つの経路バイトを含む経路フィールドを有するヘッダを含み、前記経路フィールドが前記各パケットが前記ネットワークを伝わる経路を集合的に指定し、各個々の前記経路バイトが前記各パケットが対応する前記クロスポイント・スイッチの１つを横断する経路を定義し、前記結果的経路のコピー・ステップが前記結果的経路内の各連続する前記経路バイトの値を前記ヘッダ内の別々の対応する連続経路バイトにコピーするステップを含む、前記（４）記載の方法。

（６）各ネットワーク区分が前記ネットワークの異なる半分を構成する、前記（５）記載の方法。

（７）前記装置をサービス・フェーズ及び実行フェーズで動作させるステップであって、前記第１の定義ステップ及び前記規定経路記憶ステップを前記サービス・フェーズの間に実行し、前記結果的経路アクセス・ステップ、前記結果的経路コピー・ステップ及び前記パケット経路指定ステップを前記実行フェーズの間に実行する前記動作ステップを含む、前記（５）記載の方法。

（８）前記第１の定義ステップが、トポロジ・ファイル内のネットワーク装置及び相互接続データに応答して、出所ノードとしての前記の各ノードから、宛先ノードとしてのあらゆる他の使用可能な前記ノードの１つへの全ての使用可能な最短パス経路を決定するステップであって、前記トポロジ・ファイル内に含まれるある前記装置に対するデッドロック回避指示により禁止される前記装置を通過するパスを有する経路を、前記最短パス経路から除外する、前記決定ステップと、ある前記出所ノードとある前記宛先ノード間で１つの前記最短パス経路が存在する場合、前記最短パス経路を前記経路テーブルに、前記出所ノードと前記宛先ノード間の規定経路として書込むステップと、ある前記出所ノードとある前記宛先ノード間で複数の最短パス経路が存在する場合、前記最短パス経路の中から、集合的に最小の重みを有する１つの前記最短パス経路を、前記出所ノードと前記宛先ノード間の前記規定経路として選択するステップと、前記規定経路内の各リンクに対応する別々の重みを、予め定義された量だけ増分するステップとを含む、前記（５）記載

の方法。

（９）前記全ての使用可能な経路の決定ステップが、前記全ての使用可能な最短パス経路を突き止めるブレッドス・ファースト探索を実行するステップを含む、前記（８）記載の方法。

（１０）前記装置をサービス・フェーズ及び実行フェーズで動作させるステップであって、前記第１の定義ステップ及び前記規定経路記憶ステップを前記サービス・フェーズの間に実行し、前記結果的経路アクセス・ステップ、前記結果的経路コピー・ステップ及び前記パケット経路指定ステップを前記実行フェーズの間に実行する前記動作ステップを含む、前記（９）記載の方法。

（１１）各ネットワーク区分が前記ネットワークの異なる半分を構成する、前記（１０）記載の方法。

（１２）パケット・ネットワークの外部の複数のノードを集合的に相互接続するクロスポイント・スイッチの連続ステージを含む前記ネットワークを有するシステムにおいて、パケットが前記ネットワーク及び少なくとも１つの前記スイッチを介して、規定経路上を第１の前記ノードから第２の前記ノードに伝搬されるものにおいて、前記ネットワーク内における経路指定デッドロックの発生を回避する装置であって、パケットが前記複数のノード内の個々のノードから、異なる対応する前記経路を介して、前記複数のノードのあらゆる他のノードに伝搬されるように、前記ネットワークを介する複数の規定経路を定義する第１の手段であって、前記の各定義経路が少なくとも１つのリンクに伸び、第１のネットワーク区分だけに接続される第１及び第２の前記ノード間を通過するパケットが、第２のネットワーク区分に伸びるリンクを有する経路上で伝搬されないように、前記ネットワークを前記第１及び前記第２のネットワーク区分に分割するように前記規定経路を定義する、前記第１の定義手段と、全ての前記規定経路を結果の経路テーブルに記憶する手段とを含む装置。

（１３）前記第１の定義手段が、前記第１及び前記第２のネットワーク区分にそれぞれ接続される第３及び第４のノード間を通過するパケットが、前記第１及び前記第２のネットワーク区分間に伸びる少なくとも１つのリンクを有する経路上で伝搬されるように、前記複数の規定経路を定義する、前記（１２）記載の装置。

（１４）前記ネットワーク内において出所ノードから宛先ノードに経路指定されるパケットをアセンブルする間に、前記第３及び前記第４の両ノード内において、前記パケットの結果的経路を生成するために、前記経路テーブルをアクセスし、前記結果的経路を前記パケットにコピーし、前記パケットを前記結果的経路上で前記ネットワークを介して経路指定する手段を含む、前記（１３）記載の装置。

（１５）前記経路テーブルの異なる部分がダウンロードされる前記複数の各ノードに対応する別々の局所経路テーブルであって、前記の各経路テーブル部分が前記各ノードを出所ノードとして有する全ての前記規定経路を指定する、前記局所経路テーブルと、前記出所ノードから前記宛先ノードに伝搬されるパケットの前記結果的経路を生成するために、前記パケットの前記宛先ノードにもとづき、前記出所ノードの前記局所経路テーブルをアクセスする手段とを含む、前記（１４）記載の装置。

（１６）前記の各パケットが少なくとも１つの経路バイトを含む経路フィールドを有するヘッダを含み、前記経路フィールドが前記各パケットが前記ネットワークを伝わる経路を集合的に指定し、各個々の前記経路バイトが前記各パケットが対応する前記クロスポイント・スイッチの１つを横断する経路を定義し、前記結果的経路内の各連続する前記経路バイトの値を、前記ヘッダ内の別々の対応する連続経路バイトにコピーする、前記（１５）記載の装置。

（１７）各ネットワーク区分が前記ネットワークの異なる半分を構成する、前記（１６）記載の装置。

（１８）前記第１の定義手段が、トポロジ・ファイル内のネットワーク装置及び相互接続データに応答して、出所ノードとしての前記の各ノードから、宛先ノードとしてのあらゆる他の使用可能な前記ノードの１つへの全ての使用可能な最短パス経路を決定する手段であって、前記トポロジ・ファイル内に含まれるある前記装置に対するデッドロック回避指示により禁止される前記装置を通過するパスを有する経路を、前記最短パス経路から除外する、前記決定手段と、ある前記出所ノードとある前記宛先ノード間で１つの前記最短パス経路が存在する場合、前記最短パス経路を前記経路テーブルに、前記出所ノードと前記宛先ノード間の規定経路として書込む手段と、ある前記出所ノードとある前記宛先ノード間で複数の最短パス経路が存在する場合、前記最短パス経路の中から、集合的に最小の重みを有する１つの前記最短パス経路を、前記出所ノードと前記宛先ノード間の前記規定経路として選択する手段と、前記規定経路内の各リンクに対応する別々の重みを、予め定義された量だけ増分する手段とを含む、前記（１６）記載の装置。

（１９）前記システムが並列処理システムであり、前記の各ノードが別々の処理要素を含む、前記（１６）記載の装置。

（２０）前記並列処理システムが５１２個の別々の処理要素を含み、前記スイッチが３２ポート・スイッチ・ボードに編成され、前記システムが３２個のノード・スイッチ・ボード（ＮＳＢ）と１６個の中間スイッチ・ボード（ＩＳＢ）による複数のスイッチ・ボードを含み、前記ＩＳＢが、前記の各ＮＳＢ上の１６ポートのそれぞれが、異なる対応するリンクを介して、前記の各ＮＳＢ上の同一の対応するポートに接続されるように、また前記の各ＮＳＢ上の残りの１６ポートが１６個の異なる連続する前記処理要素に接続されるように、全ての前記ＮＳ

29

Ｂを集合的に相互接続する、前記（１９）記載の装置。

【００６０】

【発明の効果】以上説明したように、本発明によれば、大規模双方向マルチステージ相互接続クロスポイント・スイッチ・ベース・パケット・ネットワークにおいて、デッドロックの無い経路指定を確立する単純でコスト・パフォーマンスの良い装置及び方法が提供される。

【図面の簡単な説明】

【図１】３２個の別々の処理要素を使用する従来の並列処理システム５のハイレベル・ブロック図である。

【図２】図１に示されるシステム５を通過するパケット３００及びその構成フィールドを表す図である。

【図３】図１に示されるシステム５を構成する処理ノード１１０、及び特にこれらのノードのメモリ内に存在してパケット経路指定を達成する様々なファイル及びテーブルを示す図である。

【図４】５１２の処理要素を含み、本発明の教示を使用する並列処理システム４００のハイレベル・ブロック図である。

【図５】システム４００内に配置される中間スイッチ・ボード（ＩＳＢ）及びそれらの相互接続ノード・スイッチ・ボード（ＮＳＢ）を示し、パケット経路の例が本発明の教示により決定される。

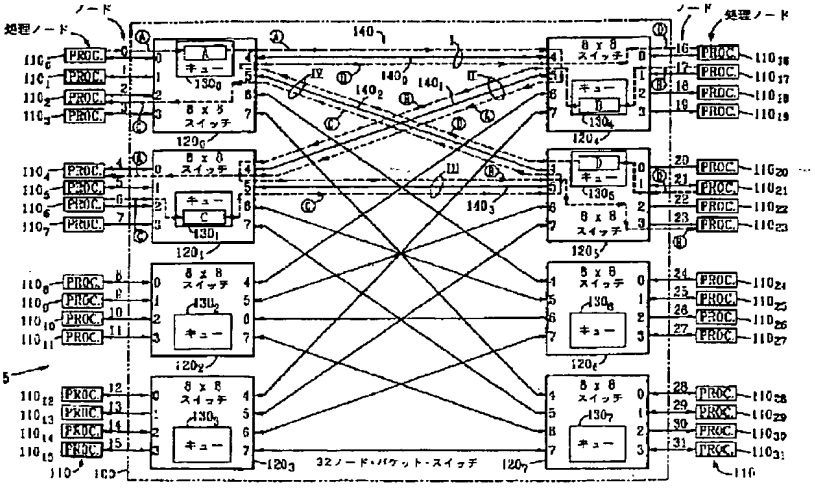
【図６】サービス・プロセッサ内で実行される経路テーブル発生器ルーチン６００のハイレベル流れ図であり、図４に示される処理要素415_{511}が、本発明の教示によりパケット経路を定義するためにシステム４００内に配置される。

30

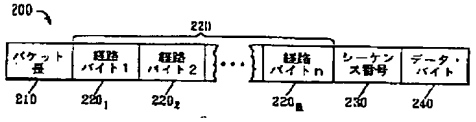
【符号の説明】

５、４００　３２プロセッサ並列処理システム
１００　ノード・パケット・スイッチ
１１０　処理要素、処理ノード
１２０　８×８双方向スイッチ回路
１４０、４５０　接続マトリックス
２００　パケット
２１０　長さフィールド
２２０　経路指定フィールド
２３０　シーケンス番号フィールド
２４０　データ・フィールド
３２０、３８０　局所経路テーブル
３４０　メモリ
３５０　トポロジ・ファイル
３６０　大域経路テーブル
３７０　初期化ルーチン
４６０　ＩＢＳ
６００　経路テーブル発生器ルーチン
６６７　否定パス

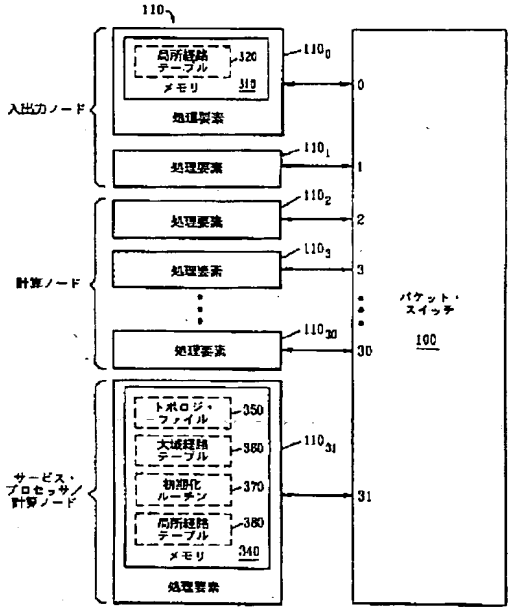
【図１】



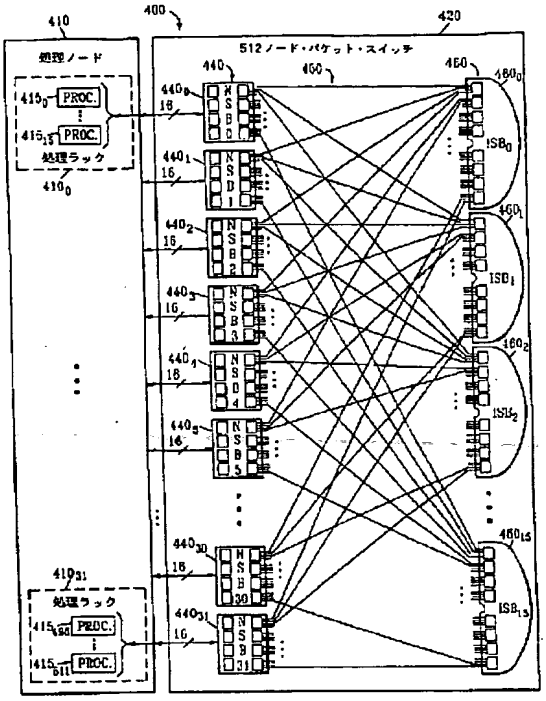
【図２】



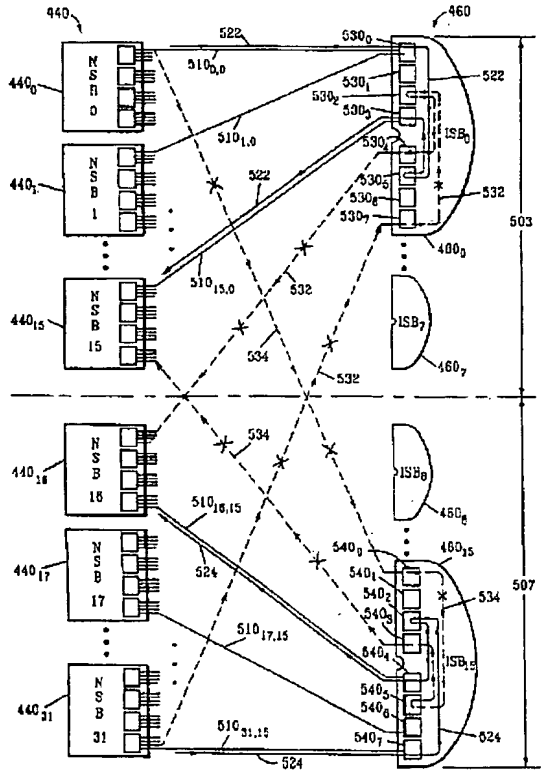
【図３】


入出力ノード
計算ノード
サービス・プロセッサ／計算ノード
局所経路テーブル
メモリ
処理要素
トポロジ・ファイル
大域経路テーブル
初期化ルーチン
パケット・スイッチ
100

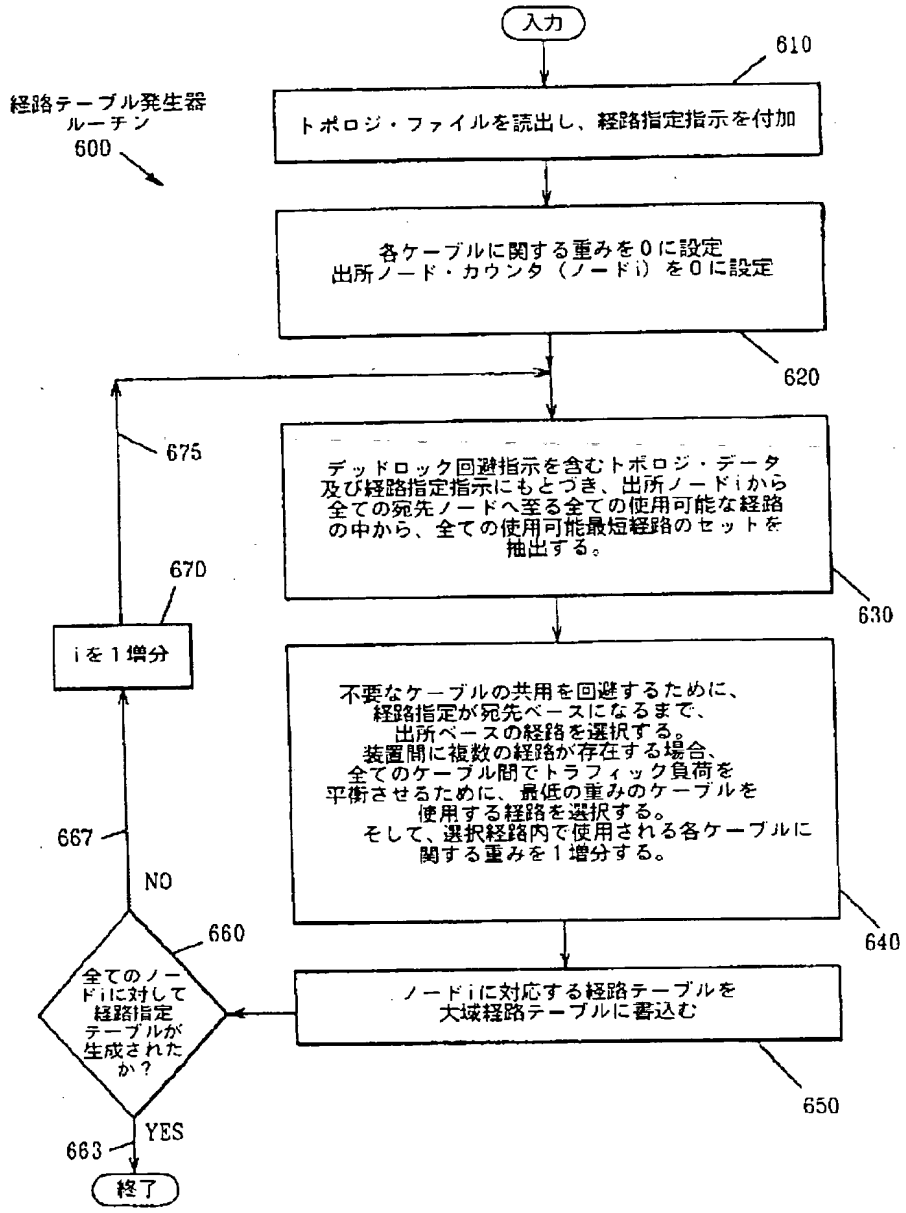
【図４】


処理ノード
処理ラック
PROC.
512ノード・パケット・スイッチ
NSB

【図５】


NSB 0
NSB 1
NSB 15
NSB 16
NSB 17
NSB 31
ISB
503
507

【図６】



フロントページの続き

(72)発明者　ロバート・フレデリック・スタッケ
アメリカ合衆国12477、ニューヨーク州ソガティーズ、リッジ・ロード　14

(72)発明者　クレイグ・ブライアン・スタンケル
アメリカ合衆国06801、コネチカット州ベスル、グリーン・パスチャー・ロード　10

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- [x] BLACK BORDERS
- [x] IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- [x] FADED TEXT OR DRAWING
- [x] BLURRED OR ILLEGIBLE TEXT OR DRAWING
- [ ] SKEWED/SLANTED IMAGES
- [x] COLOR OR BLACK AND WHITE PHOTOGRAPHS
- [ ] GRAY SCALE DOCUMENTS
- [x] LINES OR MARKS ON ORIGINAL DOCUMENT
- [ ] REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- [ ] OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.